What is Research paper?

A **research paper** is a detailed document written to communicate the findings of a study, investigation, or analysis on a specific topic. It is typically based on **original research or a review of existing research** and aims to contribute knowledge, insights, or solutions to a field of study.

Database

- Database is an organized collection of data or a type of data store based on the use of a database management system

- Small databases can be stored on a file system, while large databases are hosted on computer clusters or cloud storage.

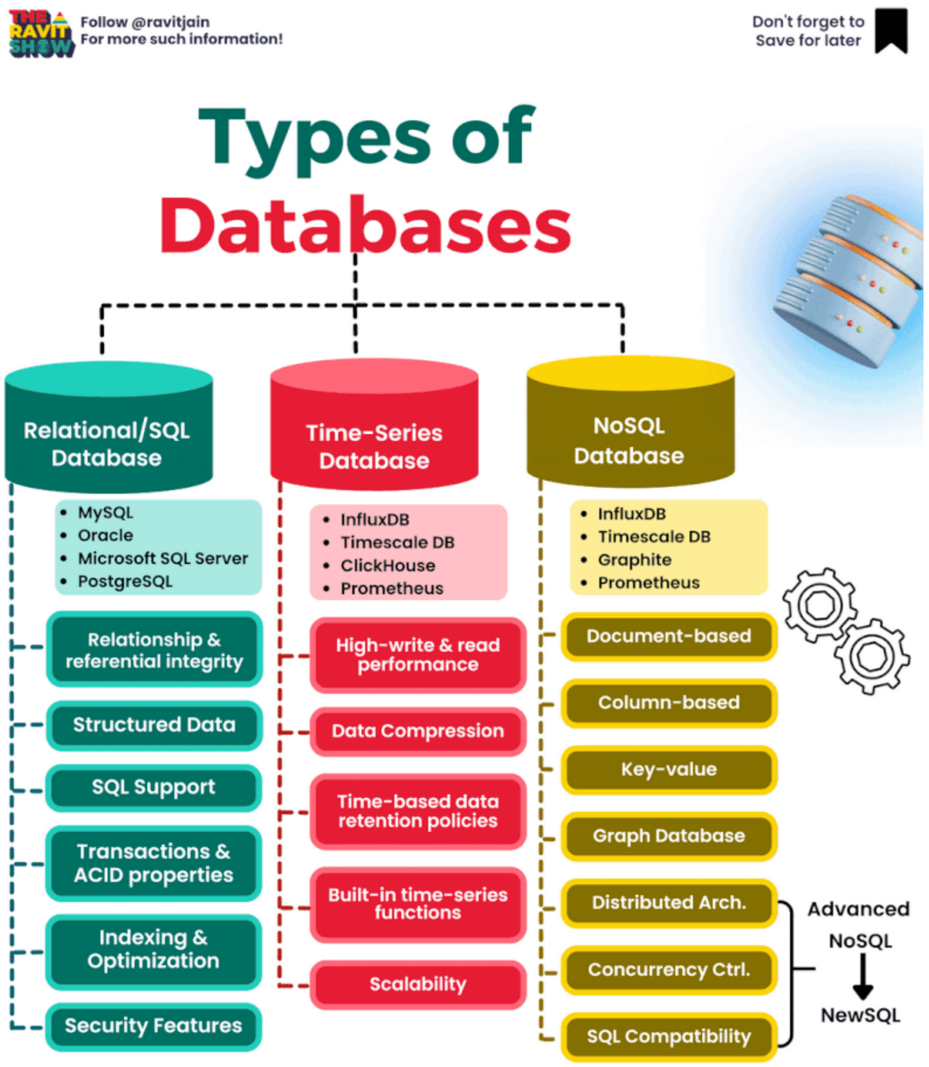
- Relational databases became dominant in the 1980s. These model data as **rows and columns** in a series of tables, and the vast majority use **SQL** for writing and querying data. In the 2000s, non-relational databases became popular, collectively referred to as **NoSQL**, because they use different **query languages**.

- **Relational databases** are globally used in most of the applications and they have good performance when they handle a limited amount of data. To handle a large volume of data like internet, multimedia and social media the use of traditional relational databases is ineffective. To overcome this problem the "NO SQL" term was introduced.

- The primary benefit of a NoSQL database is that, unlike a relational database it is **able to handle unstructured data such as documents, email, multimedia and social media efficiently.**

- Database Management Systems (DBMSs) perform four main functions:
  Data Definition: Create, change, or delete how data is organized.
  Update: Add, modify, or delete data.
  Retrieval: Find and show data based on criteria, either directly or by processing it.
  Administration: Manage users, ensure security, monitor performance, and recover data after errors.

- Database languages are specialized for specific tasks and are categorized into:
  Data Control Language (DCL): Manages access to data.
  Data Definition Language (DDL): Defines data structures like creating, changing, or deleting tables and relationships.
  Data Manipulation Language (DML): Handles data tasks like adding, updating, or removing records.
  Data Query Language (DQL): Searches for and calculates information from the database.



- There are four main strategies for storing data in **non-relational databases**:
- **Key-Value:**

  - Acts like a distributed dictionary.
  - No fixed schema (schema-less).
  - Keys can be system-generated or custom, and values can be anything (e.g., string, JSON, or BLOB).
- **Document:**
  - Examples: MongoDB.
  - Flexible, with no fixed schema.
  - Stores data in formats like JSON, BSON, XML, or BLOBs.
  - A specialized form of key-value databases but allows searching based on document content.
- **Column/Field:**
  - Examples: HBase, Hypertable.
  - Data is stored in columns grouped into logical column families.
  - Requires a predefined schema but allows generating new columns dynamically.
- **Graph-Oriented:**
  - Designed for complex data relationships.
  - Faster for certain types of queries compared to other database strategies.

- SQL combines the roles of data definition, data manipulation, and query in a single language. It was one of the first commercial languages for the relational model

- Common logical data models for databases include:
  - Navigational databases
    - Hierarchical database model
    - Network model
    - Graph database
  - Relational model
  - Entity–relationship model
    - Enhanced entity–relationship model
  - Object model
  - Document model
  - Entity–attribute–value model
  - Star schema

- *"According to the Stack Overflow Developers survey,* ***Mongo DB*** *was the most favored database for developers in the last four years. Another report* *from 3T Software Labs states that MongoDB users tried at least two different technologies, either relational or non-relational, on average in 2017. "*

- **MongoDB (**NRDBMS) Vs **RDBMS** (relational database management systems)

  MongoDB

  - MongoDB is a robust **document-oriented database** designed for ease of application, scaling, and development that supports JSON-like documents and collections. It has witnessed exponential growth in the market and is the world's fifth most popular database.

  - MongoDB is also known as a non-SQL (NoSQL) database. It stores data in the **document format model** without tables, schemas, rows, and columns. Even with heavy data loads, the document-oriented database is designed to be fast, scalable, and flexible.

  - MongoDB supports various programming languages, including C++, JavaScript, Python, PHP, Scala, Ruby, and more to the list and operating systems like **Linux, Windows, and macOS.**

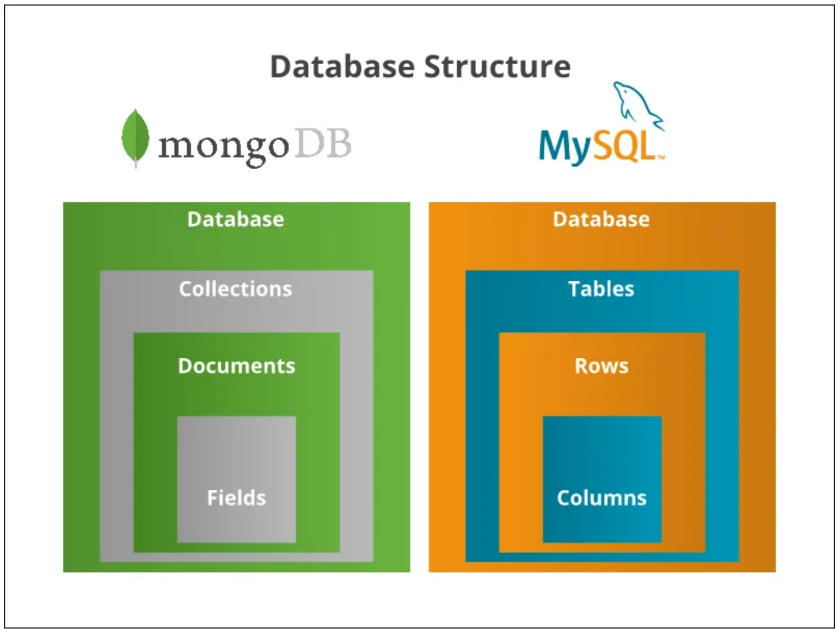
  - MongoDB is a backend data store for highly reputed companies like Twitter, Facebook, Google, IBM, Forbes, Citrix, T-Mobile, Zendesk, Sony, HTC, and many more. Some websites that use MongoDB are The New York Times, eBay, SourceForge, Craigslist, etc. According to reports by Siftery, 4,000+ organizations are using the NoSQL database for their projects

  - NoSQL database challenger MongoDB is **highly popular in the developer community**. It has 800 million monthly active users, handles 3 billion daily requests from the application, and has more than 6.5 million downloads.

  - MongoDB is also schema free i.e. a document's keys are not predefined or fixed

  - MongoDB provides **high performance, high operability, high availability, and easy scalability**. MongoDB works on fundamental idea of collection and document.

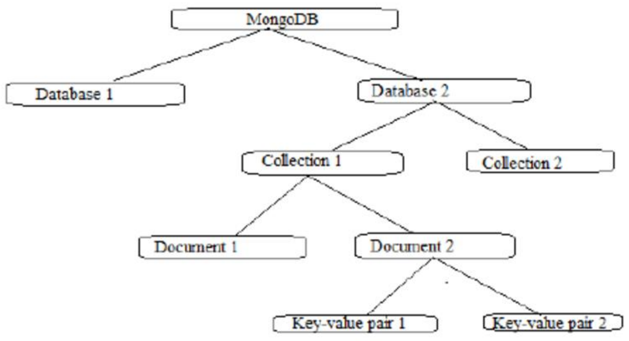
- Developers widely prefer **MongoDB** because of its ability to scale and handle projects of all sizes, making it an ideal choice for companies worldwide. From large banking systems to weekend-long hackathon projects



- Collection: - Collection is a set of MongoDB documents. It is similar to an RDBMS table. A collection operates within a single database. Collections don't enforce a schema. Documents within a collection can have many different fields. Generally, each and every document in a collection is of similar or related motive.

- Document: - A document is a set of key-value pairs. Document have dynamic schema. Dynamic schema means that the documents in the same collection don't need to have the exact same set of fields or columns or structure, and common fields in a collection's documents may hold many different types of data.

- A document is a set of fields that can be thought of as a row or tuple in a collection. It can contain complex structures like **lists, or even document**. All documents have an ID field, which is used as a primary key (field which uniquely identifies each document) and each collection can contain any type of document, but queries and indexes can only be made against one collection.



- Data Design in MongoDB database holds a **set of collections.** A collection has no pre-defined schema such as tables, and stores data as documents. BSON (objects like binary encoded JSON) are used to store documents.

- **Features of MongoDB –**

- **Schema-less Database**
  MongoDB is flexible because it doesn't require a fixed structure for data. A single collection can store different types of documents, and each document can have a different number of fields, sizes, or content.

- **Document-Oriented**
  Instead of using tables with rows and columns like in traditional databases (RDBMS), MongoDB stores data in documents. These documents use a key-value pair format, making data more flexible. Each document also has a unique ID.

- **Indexing**
  MongoDB automatically indexes fields in documents using primary and secondary indices. This makes searching data faster and more efficient. Without indexing, the database would have to scan every document, which would take much longer.

- **Scalability**
  MongoDB handles large data by **sharding**, which splits data into chunks and distributes them across multiple servers. This helps manage big data efficiently and allows adding new servers to an existing database.

- **Replication**
  MongoDB ensures data availability by creating multiple copies and storing them on different servers. If one server fails, data can still be accessed from another server.

- **Aggregation**
  Aggregation in MongoDB allows performing operations on grouped data to get a summary or computed result, similar to the **GROUP BY** function in SQL. It offers features like pipelines, map-reduce, and simple aggregation methods.

- **High Performance**
  MongoDB is fast and reliable because of features like indexing, replication, and scalability, making it a great choice for handling large amounts of data.

- **Key-value, range, geospatial, search, text search, and aggregation framework queries** are among the query types that improve the querying capabilities of the database. Additionally, MongoDB allows map-reduce queries, offering a variety of options for data retrieval. Indexes play a pivotal roleinoptimizing data access, seamlessly integrated into MongoDB without relying on external application code.

- In addition to these features, MongoDB incorporates failover mechanisms such as replica sets, ensuring data reliability. Within a replica set, a primary server handles write operations, while multiple secondary servers manage reads. An arbiter server assists during failovers without storing data, ensuring a smooth transition to the next primary server.

- Moreover, MongoDB's compatibility with various operating systemslike Windows, Linux, Mac, and Solaris ensures its accessibility across diverse platforms.

- MongoDB has its own query language named Mongo Query Language. To get certain documents from a db collection, a query document is created containing the fields that the desired documents must match. For example,

  – Insert Command
    db.users.insert ({ user id:”xyz123”, age: 34, status:”X”})

  – Select Command
    db.users.find ({ status:”X”, age: 34})

  – Delete Command
    db.users.remove ({ status:”X”})

  – Drop Command

```
db.users.drop ()
```

RDBMS

- A r*elational database management system (RDBMS)* is a regular type of database that stores and provides access to data in a **tabular** format, such as **rows and columns.** RDBMS uses Structured Query Language (SQL) to access the database.

- Modern database systems like IBM DB2, ORACLE, Microsoft Access, SQL, My-SQL, and MS SQL servers are all established based on the principles of RDBMS. The most widely used cloud-based databases are – **SQL Azure, Google Cloud SQL, IBM Db2 on Cloud, Oracle Cloud, and AWS Relational Database Service.**

- The relational database includes functions like **data accuracy, integrity, security, and consistency**.

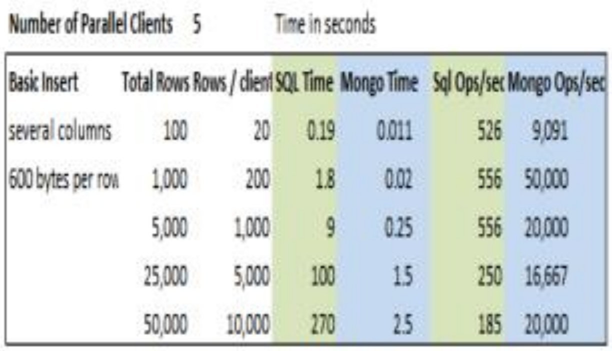
- RDBMS usually provides metadata collections and data dictionaries to handle data. Structured Query Language (SQL) is most commonly used by RDBMS to perform data-interacting tasks like creating a table and inserting data, modifying and updating data, filtering and querying data, and deleting tables or data.

- **Comparison Between MongoDB and RDBMS**

Details	**MongoDB**	**RDBMS**
Data Storage	Stores data in a document-based with no rows and columns	Stores data in a row-based table structure with fixed rows and columns
History	Developed in 2007	Developed in 1970
Hierarchical	It has inbuilt support for hierarchical data storage	It doesn’t fit hierarchical data storage

Query Language	Database vendor MongoDB supports BSON query language	Database vendor RDBMS supports SQL query language
Schema	It has a schema-less database because it doesn't need a pre-defined concept of relationship.	It usually follows the schema structure
Security	It provides security to the database	It provides robust security to the database
Foreign Key	It doesn't work with the concept of primary key - foreign key relationship	It supports foreign key
Scalability	MongoDB databases are horizontally scalable	Relational databases are vertically scalable
Principle	It follows ACID (Atomicity, Consistency, Isolation, and Durability) properties.	It follows the CAP approach (Consistency, Availability, and Partition tolerance)
Performance	RDBMS performs slowly for bulk data when compared with NoSQL database	MongoDB performs 100 times quicker than the traditional database
Joins	No complex joins are used in the database	It requires complex joins
Trigger	Triggers are not supported in NoSQL	Triggers are supported in a relational database
JavaScript	MongoDB allows JavaScript client to query	RDBMS is not suggested to access databases for JavaScript clients

- In performance testing, the authors have inserted 100 to 50,000 textbooks information into database. The cost time for MongoDB and MySQL were recorded as shown in figure. Two important factors for which MongoDB was preferred over MySQL are: Insertion Speed From the graph, we can see that MongoDB spends **less time than MySQL**, for a large amount of information as shown in figure2. It leaves **MongoDB 30× to 50× faster than MySQL** as sown in figur3.
-



Number of Parallel Clients 5 Time in seconds

Basic Insert	Total Rows	Rows / client	SQL Time	Mongo Time	Sql Ops/sec	Mongo Ops/sec
several columns	100	20	0.19	0.011	526	9,091
600 bytes per row	1,000	200	1.8	0.02	556	50,000
	5,000	1,000	9	0.25	556	20,000
	25,000	5,000	100	1.5	250	16,667
	50,000	10,000	270	2.5	185	20,000

Figure 2: INSERTION SPEED COMPARISONS

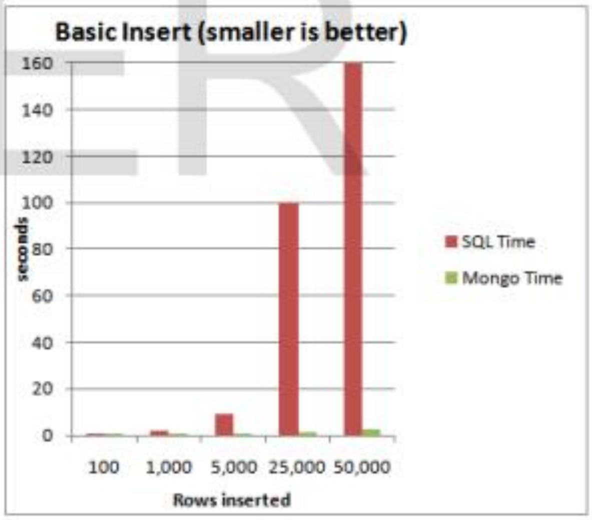


Figure 3: INSERTION TIME FOR MySQL AND MongoDB

- ## User-friendliness: MongoDB vs. MySQL

– Developers find MongoDB to be a compelling option. Anyone with programming experience can quickly and easily understand its data storage philosophy.

- Data is stored by MongoDB in collections without a set schema. Since it stores data in a flexible manner, it is especially useful for developers who wish to use a **database to support the development of their applications** but may not be experts in databases. This flexibility is a big benefit over MySQL: relational databases work best **when the concepts of normalization, referential integrity, and relational database design are understood.**

- For teams developing applications that do not require all of the security features provided by relational systems, MongoDB offers a flexible developer interface. It can store documents of different schemas, including unstructured data sets. **Web applications that can serve unstructured, semi-structured, or structured data** from the same MongoDB collection without the need for structured schemas are a common example of this type of application.

- When creating relational database solutions, updating or changing existing applications that are **already integrated with a relational system, or designing solutions for users with a lot of experience with traditional SQL scripting, MySQL is frequently selected**. Applications requiring intricate yet strict data structures and database schemas spanning numerous tables might benefit more from relational databases.

- An application used in banking that needs to maintain precise point-in-time data integrity and very strong referential integrity as well as transactional guarantees is a common example of such a system. It should be made clear, though, that **MongoDB also supports the ACID (atomicity, consistency, isolation, and durability) characteristics of transactions**. This gives developers more freedom to create a transactional data model that scales horizontally in a distributed setting without affecting the speedof multi- document transactions.

• b. Security: MongoDB vs. MySQL

- MongoDB uses a flexible permission structure in conjunction with the widely used role-based access control model. Once a user is assigned a role, that **role gives them access to particular datasets and database functions**. Every communication is secured by TLS, and **data at rest can be encrypted by writing encrypted documents to MongoDB data collections with a master key that MongoDB never has access to.**
- MySQL has an identical authentication mechanism to MongoDB and supports the same encryption features. Along with roles, **users can also be granted privileges, which grant them control over specific database operations and the ability to work with specific datasets.**